# HP OpenView

# Storage Mirroring application notes

## Data protection for Volume Shadow Copy Service

Part number: T2558–88024
First edition: April 2005

**Legal and notice information**

Hewlett-Packard Company makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Storage Mirroring application notes: Data protection for Volume Shadow Copy Service

# Introduction

Volume Shadow Copy Service (VSS) on Microsoft Windows Server 2003 allows a snapshot (or *shadow copy*) of all files on a volume to be made at regular, scheduled intervals. Shadow copies are useful for recovery of lost or corrupted files on a shared resource without the need for administrative intervention. However, shadow copies do not replace backups. If the drive that holds the shadow copies fails, the shadow copies are lost.

Storage Mirroring provides real-time enterprise data protection and replication. It provides disaster recovery and high availability for the data, thus enhancing the functionality of VSS.

This document describes how Storage Mirroring can be used with VSS on Microsoft Windows Server 2003 to offer user-initiated file restoration, in addition to the standard protection that Storage Mirroring offers for production data and applications. When VSS is enabled on both the source and target, VSS takes snapshots of the same data on both the source and target. In the event of failover, previous versions of the same data are still available on the target.

For more information about data protection using Storage Mirroring, see *HP OpenView Storage Mirroring Backup Enhancement application notes*, available from the Storage Mirroring web site:

**NOTE:** Due to the complexities of these applications, this document is intended for network administrators with experience installing, configuring, and maintaining network applications including Storage Mirroring and VSS.

# Requirements

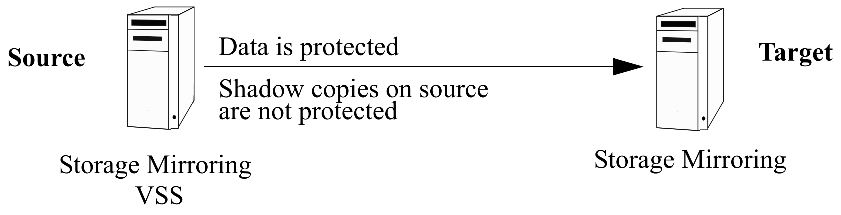
Each server must meet the following requirements:

- A licensed copy of Microsoft Windows Server 2003
- A licensed copy of Storage Mirroring version 4.4 or later

# Configurations

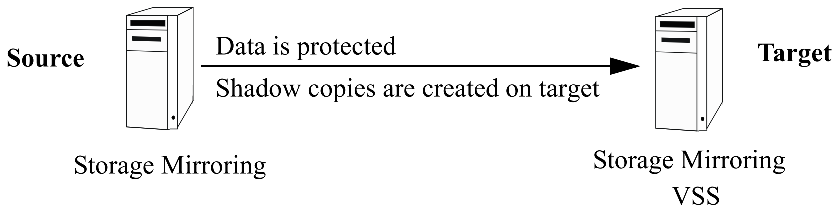
The following configurations provide options for ways to enhance VSS using Storage Mirroring.

- **VSS on Source Server**—Storage Mirroring provides real-time protection of production data and provides high availability for production applications. VSS is enabled on the source machine to provide user-initiated file recovery on the source server. However, Storage Mirroring does not provide protection for VSS snapshots.

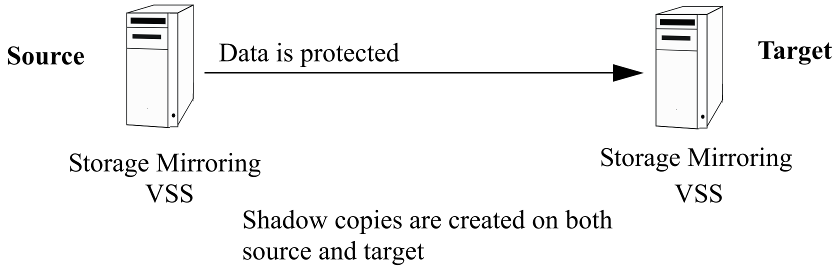


- **VSS on Target Server**—Storage Mirroring provides real-time protection of production data and provides high availability for production applications. VSS creates shadow copies of the replicated data on the target server. Creating shadow copies only on the target relieves the production server of the overhead associated with VSS.

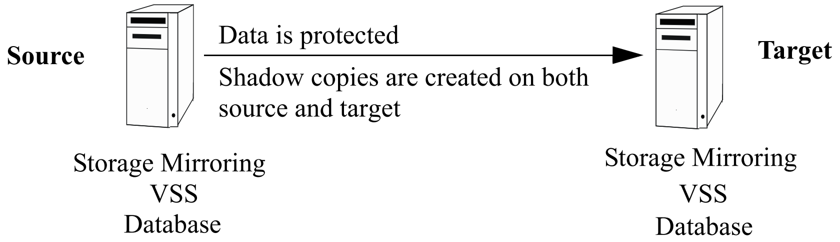
  In this configuration, the users would not have access to restoring files; however, at a user's request, the administrator can use the snapshots on the target to restore a user's file(s).



- **VSS on Source and Target**—Storage Mirroring provides failover of the source server identity and maintains a real-time copy of the source production data. VSS is enabled on the source and target servers, and snapshots are created on each server of the same data. In case of a source failure, the target can fill in for the source completely, including allowing the user complete access to production data and applications, as well as full capabilities to initiate file-level restore.



- **VSS with Transactional Applications**—Identical to the *VSS on Source and Target* configuration. However, if the data being protected is from a transactional database application (for example, SQL Server), you must ensure that the snapshot is taken at a point where the data is consistent. This is accomplished through the Storage Mirroring in-band control feature.



# Protecting data using Storage Mirroring with VSS

Prior to performing these steps, prepare your source and target machines:

- Ensure that Storage Mirroring is installed on the source and target machines. For more information, see the *HP OpenView Storage Mirroring user's guide*.
- Ensure that VSS is enabled on the source and/or target as appropriate for your configuration.
- To protect your data using Storage Mirroring with VSS, you will:
  - Create the replication set and connect to the target, as described in the next section.
  - Set up a snapshot strategy suitable for your environment, as described in "Set up snapshot strategy" on page 5.

# Create a replication set and connect to target

The following steps apply to all of the configurations identified in "Configurations" on page 3. Complete these steps to create the Storage Mirroring replication set and establish the connection between the source and target.

1. On the source, select **Start, Programs, Storage Mirroring, Management Console**.
2. Double-click your source machine to log on.
3. If database files are being replicated, enable Block Checksum All Files. If you are not replicating database files, continue with step 4.
   a. Right-click the source and select **Properties**.
   b. On the Source tab, enable **Block Checksum All Files on a Difference Mirror** and click **OK**.
4. Right-click the source machine and select **New, Replication Set.** Enter the desired name for the replication set.
5. Select the portion of the VSS volume you wish to protect (for instance, select the entire volume for a file server, or select only the data files for a database application).
6. Deselect the `System Volume Information` folder.
7. Right-click the replication set name and select **Save** to save the replication set.
8. Drag and drop the replication set onto the target. The Connection Manager dialog box opens.
9. The S**ource Server**, **Target Server**, **Replication Set,** and **Route** fields will automatically be populated. If you have multiple IP addresses on your target, verify the **Route** field is set to the correct network path. (For detailed information on connecting a source and target, see the *HP OpenView Storage Mirroring user's guide.*)
10. Select **One to One** to map the replication set data from the source to an identical volume/directory structure on the target.
11. If data from a transactional application such as a database is being protected on the source, NSI Software recommends moving or deleting orphan files on the target. If desired, configure the connection so that orphan files on the target are deleted or moved. See the *HP OpenView Storage Mirroring user's guide* for detailed instructions on orphan files.
12. Click **Connect** to start the mirror and replication processes.

Your data is protected after the mirror is complete and the Mirror Status has changed to **Idle**.

# Set up snapshot strategy

If you are using either the *VSS on Source and Target* or the *VSS with Transactional Applications* configuration, you can configure Storage Mirroring and VSS to create a data protection/snapshot solution. For more information about each of these configurations, see "Configurations" on page 3.

From the following options, select the snapshot strategy that best fits your data availability needs:

- **Asynchronous Snapshots**—Using this strategy, VSS takes snapshots of the data being protected on both the source and target servers, but the data contained in each snapshot may not match exactly due to the fact that recent data changes on the source may not have had time to replicate to the target. This strategy is useful for protecting non-database data where minor differences in the version of the files being protected would not be significant (for example, protecting a public shared folder on a file server where multiple people access the same share).

  An advantage of using this strategy is that the built-in snapshot scheduling mechanism in VSS can be used without any modifications.

  For more information about implementing this strategy, see "Setting up asynchronous snapshots" on page 6.

**NOTE:** This strategy is not recommended for databases.

- **Synchronous Snapshots**—Using this strategy, a snapshot is taken on the source machine, and a snapshot is taken on the target of the source data at the same given point. Unlike asynchronous snapshots, this strategy produces snapshots of the exact same data on both the source and target. To use synchronous snapshots, you need to use Storage Mirroring's in-band control feature to kick off snapshots (as opposed to using the VSS built-in scheduling feature). Synchronous snapshot strategies work well for transactional applications, such as databases.

  There are two options for creating synchronous snapshots:

  - **Crash-Consistent Synchronous Snapshots**—A crash-consistent state means that at any given moment, the target represents a single point in time from the source, as it might be if the source machine was powered off or crashed abruptly during operation. Unlike data-consistent snapshots, crash-consistent snapshots do not require the database application on the source machine to be stopped at any point, thus minimizing downtime.

    For more information about implementing this strategy, see "Setting up crash-consistent synchronous snapshots" on page 8.

  - **Data-Consistent Synchronous Snapshots**—A data-consistent state means that the target represents a known quiesced file state on the source, such as when an application is temporarily shut down. It typically takes less time to recover applications from a data-consistent state. This strategy is identical to the *Crash-Consistent Synchronous Snapshots* strategy, except that the database application is stopped momentarily in order to get the data in a consistent state.

    For more information about implementing this strategy, see "Setting up data-consistent synchronous snapshots" on page 10.

## Setting up asynchronous snapshots

When using the asynchronous snapshot strategy, no changes need to be made to the snapshot schedule on the source. On the target, snapshots either have to be made manually or be initiated by a script from Windows Task Scheduler. Either way, the target will have to be paused by Storage Mirroring to stop replication from occurring while the snapshot is being taken, then resumed after the snapshot is taken.

**NOTE:** The command to perform the snapshot itself using VSS is:

```
vssadmin create shadow /for=X:
```

where *X:* is the drive where the snapshot is made.

The following instructions contain step-by-step procedures for incorporating the DTCL commands used in the Text Client into a DTCL script which will allow you to automate the pause and resume process from a batch file.

**NOTE:** Pausing and resuming the target can be manually initiated through the Management Console or Text Client, or you can use DTCL commands to script it. For complete details on using the Management Console or Text Client, see the *HP OpenView Storage Mirroring user's guide*.

1. Prior to the snapshot process starting, you will want to pause execution of operations on the target. To do this, create a batch file called `presnapshot.txt` using the following sample file. Save this file to the location where Storage Mirroring is installed.

**PRESNAPSHOT.TXT**

```
rem Sample script to pause execution of operations on the target prior to starting a
snapshot.
rem Substitute the name of your target machine, username, password, and domain in the
variable
rem definitions below.
$TheTarget = "name";
$User = "username";
$Pass = "password";
$TheDomain = "domain";

login $TheTarget $User $Pass $TheDomain;
target $TheTarget;
pausetarget $TheTarget;
```

**NOTE:** Because the following files use the `login` command, which displays the password of the user ID specified, you may not want to use the network administrator account. You can create a user account specifically for this purpose, add it to the Storage Mirroring Admin security group, and grant it the minimal access necessary to complete this task.

2. Create a batch file to run this script using the following sample `PRESNAPSHOT.BAT` batch file.

**PRESNAPSHOT.BAT**

```
rem Sample batch file to run the presnapshot.txt script.

C:
cd c:\program files\Storage Mirroring
cmd /c DTCL -f "c:\program files\Storage Mirroring\presnapshot.txt"
```

3. After the snapshot process is complete, you will want to resume execution of operations on the target. To do this, create a batch file called `postsnapshot.txt` using the following sample file. Save this file to the location where Storage Mirroring is installed.

**POSTSNAPSHOT.TXT**

```
rem Sample script to resume execution of operations on the target after the snapshot is
complete.
rem Substitute the name of your target machine, username, password, and domain in the
variable
rem definitions below.
$TheTarget = "name";
$User = "username";
$Pass = "password";
$TheDomain = "domain";

login $TheTarget $User $Pass $TheDomain;
target $TheTarget;
resumetarget $TheTarget;
```

4. Create a batch file to run this script using the following `POSTSNAPSHOT.BAT` sample batch file.

**POSTSNAPSHOT.BAT**

```
rem Sample batch file to run the postsnapshot.txt script.


C:
cd c:\program files\Storage Mirroring
cmd /c DTCL -f "c:\program files\Storage Mirroring\postsnapshot.txt"
```

5. Run `presnapshot.bat` before starting your snapshot and run `postsnapshot.bat` after the snapshot is complete.

**NOTE:** You can also incorporate these two scripts into an automated script that can be scheduled to run at the desired time through Windows Task Scheduler.

Depending on the length of time required to complete your snapshot, Storage Mirroring may not be able to queue all of the replication data. If the queue has reached the maximum size limit, Storage Mirroring will automatically disconnect the connections and attempt to reconnect them. This is called an auto-disconnect. If you are experiencing frequent auto-disconnects because the queues have reached the maximum limit while the snapshot is processing, you can:

- Increase the amount of disk space on the volume where the Storage Mirroring queue is located or move the extended queue to a larger volume
- Disable auto-reconnect and reconnect manually or in a post-snapshot DTCL script (It may also be desirable to script a DTCL disconnect command in a pre-snapshot script and reconnect in the post-snapshot script.)
- Create a DTCL script to disconnect Storage Mirroring before the snapshot and reconnect and remirror after the snapshot is complete

See the *HP OpenView Storage Mirroring user's guide* for additional details.

## Setting up crash-consistent synchronous snapshots

For this strategy to work, you will have to initiate snapshots on both the source and target using Storage Mirroring's in-band control feature, or use the Windows Task Scheduler to accomplish this. By using Storage Mirroring to insert and run tasks at various points during the replication of data, you can coordinate a point-in-time snapshot with real-time replication.

In order to synchronize snapshots on the source and target machines so that they contain the same data, the process to create VSS snapshots on both the source and target servers is initiated via the same task command. Here is how the process would work.

1. Storage Mirroring and an application are both running on the source. Only Storage Mirroring is running on the target.
2. The application data is changing on the source and Storage Mirroring is capturing those data changes and transmitting them to the target.
3. A script is launched that initiates a Storage Mirroring task command.
4. The Storage Mirroring task command is processed immediately on the source, initiating snapshot creation on the source server. Then, the command is transmitted inline with the source replication data to the target.
5. The data is applied on the target as it is received. Since the task command was inserted inline, the replication data from the source is applied to the target first. When the target gets to the Storage Mirroring task command, the target data will be in the exact same state as the source data when the task command was processed by the source.
6. The target processes the Storage Mirroring task command and initiates snapshot creation on the target. Since the Storage Mirroring task command is user-defined, you can insert any valid executable or batch file.

The following batch files outline an example of inserting a task command to initiate synchronized VSS snapshots on the source and target servers. You should modify the batch files to accommodate your own configuration.

**NOTE:** The `dosnapshot.bat`, `task.txt`, and `snapshotcommand.bat` files should be stored on the source in the Storage Mirroring directory (typically `c:\Program Files\Storage Mirroring`). The `snapshotcommand.bat` file should also be stored in the Storage Mirroring directory on the target. The files may be run manually when desired, or they can be scheduled using the Windows Task Scheduler. See your Microsoft Windows reference guide for details on scheduling.

Before these batch files can be executed, Storage Mirroring task command processing must be enabled.

1. In the Storage Mirroring Management Console, double-click on the source server to log in.
2. Right-click on the server and choose **Properties**.
3. Click on the Setup tab. Select **Enable Task Command Processing**.
4. Repeat steps 1-3 on the target server.

For complete details on the DTCL `queuetask` command that is used, see the *HP OpenView Storage Mirroring user's guide*. For complete details on Microsoft commands, see your Microsoft reference guide.

The following are samples of batch files called in this process.

### DOSNAPSHOT.BAT

```
REM The following file is a sample batch file that runs the Storage Mirroring Command Line
Client File Entry.


C:
cd c:\Program Files\Storage Mirroring
cmd /c DTCL -f "c:\Program Files\Storage Mirroring\task.txt"
```

### TASK.TXT file used in DOSNAPSHOT.BAT

```
#The following is a sample DTCL file that logs in to a source and target server and inserts task #
#commands that will trigger a snapshot on both servers. #

#Substitute the name of your source and target machines as well as the login credentials. If you do not #
#want the login credentials of the administrator account exposed in this file, you can use another #
#account, as long as it is a member of the Storage Mirroring Admin security group on both the source #
#and target servers. #

$TheSource = "SourceName";
$TheTarget = "TargetName";
$TheUserName = "UserName";
$ThePassword = "Password";
$TheDomain = "DomainName";
login $TheSource $TheUserName $ThePassword $TheDomainName;
login $TheTarget $TheUserName $ThePassword $TheDomainName;
source $TheSource;
queuetask backup_process to $TheTarget onqueue=SnapshotCommand.bat onexecute=SnapshotCommand.bat
timeout=forever;
```

**SNAPSHOTCOMMAND.BAT file used in TASK.TXT**

```
REM The following is a sample batch file to perform a snapshot. In this command, X: is the
drive where the REM snapshot is made.


vssadmin create shadow /for=X:
```

## Setting up data-consistent synchronous snapshots

For this strategy to work, you will have to initiate snapshots manually on both the source and target using Storage Mirroring's in-band control feature, or use the Windows Task Scheduler to accomplish this.

To ensure that all data is committed to the database and that the data is in a data-consistent state, you will need to stop the database service. With task command processing, you can stop the source service just long enough to identify that stopped point in time as a data-consistent state, insert a task at that point into the Storage Mirroring replication queue to trigger a snapshot on the source and/or target, and then restart the service. Here is how the process would work.

1. Storage Mirroring and an application are both running on the source. Only Storage Mirroring is running on the target.
2. The application data is changing on the source and Storage Mirroring is capturing those data changes and transmitting them to the target.
3. A script is launched (either manually or perhaps scheduled by the Windows Task Scheduler) that stops the application service on the source, pauses to give the service time to shutdown and write the data to disk, initiates a Storage Mirroring task command, and then restarts the application service on the source.
4. The Storage Mirroring task command is processed immediately on the source, initiating snapshot creation on the source server. Then, the command is transmitted inline with the source replication data to the target.
5. The data is applied on the target as it is received. Since the task command was inserted inline, the replication data from the source is applied to the target first. When the target gets to the Storage Mirroring task command, the target data will be in the exact same state as the source data when the source application service was stopped. Since this was a stable point on the source, it is also a stable point on the target.
6. The target processes the Storage Mirroring task command and completes whatever task is defined. Since the Storage Mirroring task command is user-defined, you can insert any valid executable or batch file.

The following batch files outline an example of stopping a Microsoft SQL Server 2000 database, inserting a task command to initiate VSS snapshot on the target, and then restarting the SQL database. You can modify the batch files to accommodate your own application(s) and/or snapshot solution(s).

The `sql_snapshot.bat`, `task.txt`, and `snapshotcommand.bat` files should be stored on the source in the Storage Mirroring directory (typically `C:\Program Files\Storage Mirroring`). The snapshotcommand.bat file should also be stored in the Storage Mirroring directory on the target. The files may be run manually when desired, or they can be scheduled using the Windows Task Scheduler. See your Microsoft Windows reference guide for details on scheduling.

Before these batch files can be executed, Storage Mirroring task command processing must be enabled.

1. In the Storage Mirroring Management Console, double-click on the source server to log in.
2. Right-click on the server and choose **Properties**.
3. Click on the Setup tab. Select **Enable Task Command Processing**.
4. Repeat steps 1-3 on the target server.

For complete details on the DTCL `queuetask` command that is used, see the *HP OpenView Storage Mirroring user's guide*. For complete details on Microsoft commands, see your Microsoft reference guide.

The following are samples of batch files called in this process.

### SQL_SNAPSHOT.BAT

```
REM The following file is a sample batch file that stops the Microsoft SQL Server 2000 services on the
REM source, pauses to allow the source to write all of the SQL data to the source, inserts a Storage
Mirroring
REM task command into the Storage Mirroring replication process, and then restarts the SQL services. This
REM batch file should be stored on the source server.

REM Storage Mirroring task command processing must be enabled and there must be an active Storage Mirroring
REM connection for this process to function properly. See the Storage Mirroring User’s Guide for assistance
REM in enabling task command processing and establishing a connection.

REM The following lines stop the Microsoft SQL Server 2000 services without requiring administrator
REM interaction.

net stop "Distributed Transaction Coordinator"
net stop "Message Queuing"
net stop "MSSQLServer" /y
net stop "SQLServerAgent"

REM The following line pauses the execution of this batch file for 120 seconds (2 minutes) so that any
REM remaining application data can be written to disk on the source. This command is available from the
REM Windows 2000/NT Resource Kit. If you do not have the Resource Kit, you will need to determine
REM another method to delay script processing. You may need to adjust the setting to accomodate the
REM amount of data your application is processing and the speed of your environment.

sleep 120

REM Since the source service is now stopped on the source and all of the data has been written to disk,
REM the application is now in a stable state. When the target reaches this exact point, you want to
REM initiate the snapshot. The following lines insert a Storage Mirroring task command to initiate a
snapshot
REM on the target.

C:
cd c:\Program Files\Storage Mirroring
cmd /c DTCL -f "c:\Program Files\Storage Mirroring\task.txt"

REM Now that the command to perform the snapshot been inserted inline with the data, the service can be
REM restarted. New updated data will fall inline behind the task command that was just inserted.

REM The following lines start the Microsoft SQL Server 2000 services on the source.

net start "Distributed Transaction Coordinator"
net start "Message Queuing"
net start "MSSQLServer"
net start "SQLServerAgent"
```

The following are samples of batch files called in this process.

### TASK.TXT file used in SQL_SNAPSHOT.BAT

```
# The following is a sample DTCL file that logs into a source and target server and inserts task #
# commands that will trigger a snapshot on both servers. #


# Substitute the name of your source and target machines as well as the login credentials. If you do #
# not want the login credentials of the administrator account exposed in this file, you can use another #
# account, as long as it is a member of the Storage Mirroring Admin security group on both the source #
# and target servers. #


$TheSource = "SourceName";
$TheTarget = "TargetName";
$TheUserName = "Administrator";
$ThePassword = "Password";
$TheDomain = "DomainName";
login $TheSource $TheUserName $ThePassword $TheDomainName;
login $TheTarget $TheUserName $ThePassword $TheDomainName;
source $TheSource;
queuetask backup_process to $TheTarget onqueue=SnapshotCommand.bat onexecute=SnapshotCommand.bat
timeout=forever;
```

### SNAPSHOTCOMMAND.BAT file used in TASK.TXT

```
REM The following is a sample batch file to perform a snapshot. In this command, X: is the
drive where the REM snapshot is made.


vssadmin create shadow /for=X:
```